

Homework 1

CS350 2019-20 Semester 1

Due Date: Sunday, August 25 2019

Instructions. You must turn in your submissions individually. The code can be submitted in a single file named `<rollno>.hw1.oz` Please acknowledge all your sources, for each question, in a README file. If you discuss with your friends, please acknowledge them for each question which you have discussed with them.

- Write code for the following Oz functions in the declarative sequential model. Please make sure that you define the boundary cases and error cases appropriately.
 - The function `{Drop N Xs}` will return all but the first `N` elements of the list `Xs`. **[10 points]**
 - The function `{Zip Xs Ys}` takes as input two lists of equal length, say `N`, and returns a list of pairs where the i^{th} pair, $1 \leq i \leq N$, consists of the i^{th} element of `Xs` followed by the i^{th} element of `Ys`. **[10 points]**
 - The function `{DeDup Xs}` eliminates consecutive occurrences of the same element with a unique occurrence. For example, `{DeDup [1 1 2 2 3 3 3]}` should return `[1 2 3]`. **[10 points]**
 - Redefine `{Length Xs}` using `Map` and `FoldR`. **[10 points]**
 - Redefine `Map` using `FoldR`. **[10 points]**
 - Using the binary tree format in the notes, define a function `{MapTree F T}` which outputs a binary tree where every element of the output tree is obtained by applying `F` to the corresponding element in the input tree. **[10 points]**
- Write code for an Oz function `{Subsets Xs}` which returns a list of all the subsets of the set of elements in `Xs`. You can assume that the elements of `Xs` are unique. Each subset should be represented as a list. The subsets may be listed in any order. The code should be written in the declarative sequential style.
For example, `{Subsets [1 2 3]}` may return `[[[]] [1] [2] [3] [1 2] [2 3] [1 2 3]]`. **[15 points]**
- Write a lazy Oz function `{LFilter Predicate Xs}` where `Predicate` is a unary function which returns `true` or `false` and `Xs` is a list of elements. The return value should be the sublist of elements from `Xs` for which `Predicate` evaluates to `true`.
For example, `{LFilter fun {$ X} X>0 end [1 0 1 2]}` should return `[1 1 2]`. **[10 points]**
 - Using `LFilter`, write a lazy Oz function which computes the list of primes, using the Sieve of Eratosthenes. **[15 points]**

Errata

1. (August 13, 2019.) In Q3a, the return value in the example was originally given as $[1 \ 2]$. Thanks to Bhavishya for pointing out this error.